

Amendments to the Specification:

Please insert the following paragraph beginning at page 1,
line 2:

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is a continuation application of and
claims priority to U.S. Patent Application Serial No.
09/472,545, filed December 27, 1999, now U.S. Patent No.
6,724,948.

Please replace the paragraph beginning at page 6, line 13,
with the following amended paragraph:

In the first pass 18, image 22 is read from memory 13 at an
image update rate, e.g., 60 frames per second (60Hz). As shown
in FIG. 5, to create a vertically scaled image 24, the first
pass 18 reads four memory locations to fetch four horizontal
pixel segments 25 (each segment having four pixels of image 22).
Each horizontal pixel segment 25 includes 32 bits (four pixels
of eight bits each) and travels on a 32-bit data bus to a four-
tap filter where they await compression. The horizontal pixel
segments 25 may vary in size, e.g., eight pixels of eight bits
each, to fit on a different sized bus, e.g., 64 bits. With the

four horizontal pixel segments 25 stored in the four-tap filter, the first pass 18 compresses four vertical pixel segments 29 (formed from four horizontal pixel segments 25) at the image update rate to form four pixels of vertically scaled image 24. The first pass 18 then proceeds to process the remaining horizontal pixel segments 25 from top to bottom, working from the left column to the right column.

Please replace the paragraph beginning at page 8, line 20, with the following amended paragraph:

An input pixel formatting block (IPFB) 32 requests horizontal pixel segments of the image 22 from memory interface 30, generates the addresses required to gather the horizontal pixel segments in the proper sequence, formats them to the expected form, and sends them to the second block, a pixel filtering block (PFB) 34. The PFB 34 filters the vertical pixel segments formed from the horizontal pixel segments as it receives them from the IPFB 32. This filtering affects ~~effects~~ the vertical scaling, thus making the PFB 34 the key block in the hardware vertical scaler 40. After filtering the pixels, the PFB 34 outputs them to the third block, an output pixel formatting block (OPFB) 36. The OPFB 36 receives pixels from

the PFB 34, collects them for rendering back to memory interface 30, and generates the addresses required to assemble the vertically scaled image 24 in memory 30.

Please replace the paragraph beginning at page 11, line 5, with the following amended paragraph:

Therefore, in general what the IPFB 32 does is retrieve horizontal pixel segments 25 (each segment having four pixels for a 4:1 downscaling) of image 22 from memory and present them to the PFB 34 for vertical scaling.

Please replace the paragraph beginning at page 11, line 12, with the following amended paragraph:

As shown in FIGS. 14 and 15, the PFB 34 starts its operations when it sees an Off to On transition, triggered by x_Done 74, on one of a Y_Done 80, U_Done 82, or V_Done 84 input. A filter datapath 35 in the PFB 34 fetches two horizontal pixel segments 25 (each segment has one quad of data) at a time from the IPFB 32. The horizontal pixel segments 25 are fetched from top to bottom, working from the left column to the right column. As the PFB 34 reads horizontal pixel segments 25, it filters them based on a scale factor. The PFB 34 can properly operate

on both luminance and chrominance (subsampled) pixels. The PFB 34 outputs the number of filtered pixels to form the same sized pixel segment as the PFB received to an output formatting block (OPFB) 36.

Please replace the paragraph beginning at page 12, line 1, with the following amended paragraph:

Therefore, in general what the PFB 34 does is vertically scale pixel segments (each segment having four pixels for a 4:1 downscaling) of the original image 22 and output the scaled pixels to the OPFB 36.